1000-9825/2003/14(02)0247 ©2003 Journal of Software 软 件 学 报 Vol.14, No.2

具有全序时态类型集时态函数依赖集的研究*

姚春龙[1,2+], 郝忠孝[1,2,3]

[1](哈尔滨工业大学 计算机科学与工程系,黑龙江 哈尔滨 150001)
[2](齐齐哈尔大学 计算机科学与技术系,黑龙江 齐齐哈尔 161006)
[3](哈尔滨理工大学 计算机与控制学院,黑龙江 哈尔滨 150080)

Study on Set of Temporal Functional Dependencies with Totally Ordered Set of Temporal Types

YAO Chun-Long[1,2+], HAO Zhong-Xiao[1,2,3]

[1](Department of Computer Science and Engineering, Harbin Institute of Technology, Harbin 150001, China)
[2](Department of Computer Science and Technology, Qiqihr University, Qiqihar 161006, China)
[3](School of Computer and Control, Harbin University of Science and Technology, Harbin 150080, China)
+ Corresponding author: Phn: 86-452-2738171, Fax: 86-452-2725454, E-mail: yclhw@263.net
http://www.hit.edu.cn
Received 2001-07-28; Accepted 2001-11-05

Yao CL, Hao ZX. Study on set of temporal functional dependencies with totally ordered set of temporal types. *Journal of Software*, 2003,14(2):247~252.

Abstract: The purpose of good database logical design is to eliminate data redundancy and insertion, deletion and update anomalies. For temporal databases, temporal schemes may be normalized by using constraints of temporal functional dependencies (TFDs) with multiple temporal granularities. However, the adoption of temporal dimension and usage of multiple temporal granularities make it very complicated to design a temporal database. Generally, the set of temporal types that can be processed by a system and involved in lots of applications, meet totally ordered relation, and the set of TFDs with a totally ordered set of temporal types is closely related to the Armstrong axioms of traditional functional dependencies (FDs). By analyzing the existing relationships between TFDs and FDs and utilizing corresponding algorithms for traditional set of FDs, some important algorithms such as membership and finite closure of attributes algorithms are proposed for given set of TFDs. These algorithms are the basis of further normalization for temporal databases.
Key words: temporal database; functional dependency (FD); normalization; relational scheme; logical design; logical implication

摘 要: 好的数据库逻辑设计目标是消除数据冗余以及插入、删除和更新异常.对于时态数据库,可以通过具有多时间粒度的时态函数依赖(TFDs)约束对时态数模式进行规范化.但是由于时间维的引入和多时间粒度的使

---

* Supported by the Natural Science Foundation of Heilongjiang Province of China under Grant No.F00-06 (黑龙江省自然科学基金)

第一作者简介: 姚春龙(1971－),男,黑龙江佳木斯人,博士生,讲师,主要研究领域为数据库系统与理论.

用而给数据库设计带来巨大的复杂性.一般来说,系统所能处理的和相当多的应用所涉及到的时态类型集满足全序关系,并且具有全序时态类型集的 TFD 集的推导规则与传统函数依赖(FDs)的 Armstrong 公理有着紧密的联系.通过分析 TFDs 与 FDs 之间存在的联系,利用传统 FD 集的相应算法,提出了成员籍、有限属性闭包等 TFD 集的一些重要算法.这些算法是时态数据库进一步规范化的基础.

关键词: 时态数据库;函数依赖(FD);规范化;关系模式;逻辑设计;逻辑蕴涵

中图法分类号: TP311 **文献标识码**: A

由于时间维的引入,时态数据库能够反映现实世界事物随时间变化的特性,但也给时态数据库的设计带来了非常大的困难.近年来,对于时态数据库设计有了相当多的研究,Jensen[1~3],Wijsen[3,4]及 Wang[5]等人提出了各自的时态数据依赖的概念,其中 Wang[5]提出的时态函数依赖(TFD)能够处理多时间粒度,较灵活地反映了现实世界.鉴于此, Wijsen[6]将其扩充到对复杂对象的约束.

Wang[5]基于 TFD 系统地讨论了具有多时间粒度的时态数据库的逻辑设计问题,定义了时态三范式(T3NF)及时态 Boyce-Codd 范式(TBCNF),并提出了相应的分解算法.由于 Wang[5]提出的算法涉及到复杂的时态类型间的运算和时态范式的判定问题,至今还没有在任何文献中发现解决这些问题的方法,因此还无法进行有效的数据库逻辑设计.类似于对传统 FD 集的处理,设法对 TFD 集进行化简和综合,从而避开时态范式的判定问题.这就要涉及到 TFD 集的成员籍和属性集闭包等问题.本文将对具有全序时态类型集的 TFD 集及相关的一些重要算法进行深入的讨论.首先介绍时态类型和 TFD 的有关概念,然后分析具有全序时态类型集的 TFD 集的推导规则与 Armstrong 公理的联系,进而研究相应的算法.

1 基本概念

有关传统函数依赖的论述见文献[7].文献[5]对于时态模式以及时态函数依赖(TFD)等有详细论述,下面描述本文所涉及到的一些基本概念.为描述时态类型,设全体实数集代表时间,记 R 为实数集,2^R 来表示 R 的幂集.

定义 1(时态类型)[5]. 时态类型是一个从确定的整数(时刻)集合到 2^R(绝对时间集合)的投影μ,使得对所有确定的整数 i 和 j,$i<j$,下列条件满足:

(1) 若$\mu(i)\neq\varnothing$且$\mu(j)\neq\varnothing$,那么每一个$\mu(i)$中的实数小于所有$\mu(j)$中的实数,且

(2) 若$\mu(i)=\varnothing$,则$\mu(j)=\varnothing$.

例 1:现实生活中可以把 Day(天)、Month(月)、Year(年)以及 Week(星期)等定义为不同的时态类型.例如可以定义 Year 从 1800 年起始,那么 Year(1)与 1800 年对应,Year(2)与 1801 年对应.

定义 2(细于关系)[5]. μ_1 和μ_2 是时态类型,如果对每一个确定的整数 i,存在整数 j 满足$\mu_1(i)\subseteq\mu_2(j)$,那么称$\mu_1$细于$\mu_2$,记作$\mu_1\preccurlyeq\mu_2$,也称$\mu_2$ 的时刻 j 覆盖μ_1 的时刻 i.

对于任何时态类型集都存在一个最小下界和最大上界,分别记作μ_{Bottom} 和μ_{Top}.它们的定义为:(1) 对每个 $i>1$,$\mu_{\text{Top}}(1)=R$,$\mu_{\text{Top}}(i)=\varnothing$;(2) 对每个 i,$\mu_{\text{Bottom}}(i)=\varnothing$.在下面的描述中,在不产生二义性的情况下,用 Top 和 Bottom 分别表示μ_{Top} 和μ_{Bottom}.对于任何一对时态类型μ_1 和μ_2,分别存在一个最大下界和最小上界,分别记作 glb(μ_1,μ_2)和 lub(μ_1,μ_2).这里我们记$\mu_1\prec\mu_2$ 表示$\mu_1\preccurlyeq\mu_2$ 且$\mu_1\neq\mu_2$.通过定义 2 可知$\mu\preccurlyeq\mu$,且如果存在$\mu_1\preccurlyeq\mu_2$,$\mu_2\preccurlyeq\mu_1$,那么$\mu_1=\mu_2$,进一步可以看到,细于关系是反对称的、传递的,由此可知,对于任何时态类型集 T,T 对于$\preccurlyeq$是偏序集.

定义 3(全序时态类型集). 给定时态类型集 T,若 T 对于细于关系是全序的,即 T 是偏序集且对于 T 中的任何两个时态类型μ 和ν,必有$\mu\preccurlyeq\nu$或$\nu\preccurlyeq\mu$,则称 T 是全序时态类型集.

定义 4(集细于关系)[5]. $\{\mu_1,\mu_2,\ldots,\mu_n\}$是一个时态类型集,$\nu$是一个时态类型,如果对每一个确定的整数 i,存在 $1\le k\le n$ 及整数 j,使得$\nu(i)\subseteq\mu_k(j)$,则称ν集细于$\{\mu_1,\mu_2,\ldots,\mu_n\}$,记作$\nu\preccurlyeq_C\{\mu_1,\mu_2,\ldots,\mu_n\}$.

利用时态类型,Wang[5]定义了具有多时间粒度的时态函数依赖(TFDs),下面是 TFD 集的一个例子.

例 2:考虑时态模式(Emp,Day),其中 Emp=$\langle$*E#*(员工号),Ename(姓名),Salary(薪水),Dept(部门),Office(办公室)$\rangle$,若有如下语义:(1) 员工的名字始终不变;(2) 员工的薪水和工作部门一个月内不能改变;(3) 每个员工一天

内只能在一个办公室工作.那么可以得到一个 TFD 集 $F=\{E\#\rightarrow_{\mathrm{Top}}\mathrm{Ename}, E\#\rightarrow_{\mathrm{Month}}\mathrm{Salary}, E\#\rightarrow_{\mathrm{Month}}\mathrm{Dept}, E\#\rightarrow_{\mathrm{Day}}\mathrm{Office}\}$.

定义 5(时态关键字)[5]. (R,μ)是一个时态模式,F 是仅包含 R 中属性的 TFD 集,属性集 $X\subseteq R$,若 $X\rightarrow_\mu R$ 被 F 逻辑蕴涵,则称 X 是(R,μ)的一个时态超关键字,若对每一个属性 $A\in X$,$X-\{A\}$都不是(R,μ)的时态超关键字,则称 X 是(R,μ)的一个时态候选关键字,简称时态码.

例 3:对于上例中的时态模式(Emp,Day),容易看出 $E\#$是(Emp,Day)的时态码.

文献[5]中给出 TFD 的有效和完备的推导规则:

TFD_1(自反规则):如果 $Y\subseteq X$,那么对每个时态类型μ,$X\rightarrow_\mu Y$;

TFD_2(增广规则):如果 $X\rightarrow_\mu Y$,那么 $XZ\rightarrow_\mu YZ$;

TFD_3(传递规则):如果 $X\rightarrow_\mu Y$ 并且 $Y\rightarrow_\mu Z$,那么 $X\rightarrow_\mu Z$;

TFD_4(继承规则):如果对于 $n\ge 1, X\rightarrow_{\mu_1} Y,\ldots,X\rightarrow_{\mu_n} Y$,那么对每个$\mu\preccurlyeq_C\{\mu_1,\mu_2,\ldots,\mu_n\}$,$X\rightarrow_\mu Y$.

由自反规则和继承规则可以看到,对给定 TFD 集 F 可能导出无穷个 TFD,为此,文献[5]中给出 3 个有效并且完备到继承性的有限推导规则:

FTFD_1(限制自反规则):如果 $Y\subseteq X$,那么 $X\rightarrow_{\mathrm{Top}} Y$;

FTFD_2(限制增广规则):如果 $X\rightarrow_\mu Y$,那么 $XZ\rightarrow_\mu YZ$;

FTFD_3(扩展传递规则):如果 $X\rightarrow_\mu Y$ 并且 $Y\rightarrow_\nu Z$,那么 $X\rightarrow_{\mathrm{glb}(\mu,\nu)} Z$.

如果 $X\rightarrow_\mu Y$ 由 F 使用上述 3 个有限推导规则导出,那么 F 有限导出 $X\rightarrow_\mu Y$,记作 $F\vdash_f X\rightarrow_\mu Y$.同时,记 $F\vdash X\rightarrow_\mu Y$ 表示 $X\rightarrow_\mu Y$ 由 F 导出,$F\vDash X\rightarrow_\mu Y$ 表示 $X\rightarrow_\mu Y$ 被 F 逻辑蕴涵[5].

为描述方便,这里给出操作$\pi_\tau(F)$:F 是 TFD 集,τ是实数集(绝对时间),$\pi_\tau(F)$是在时间τ内有效的非时态 FD(传统函数依赖),即$\pi_\tau(F)=\{X\rightarrow Y \mid \exists X\rightarrow_\nu Y\in F \text{ 且} \exists j(\tau\subseteq\nu(j))\}$.显然,$\pi_\varnothing(F)$是 F 的非时态版本.

定义 6(有限闭包)[5]. F 是一个 TFD 集,F 的有限闭包是用 FTFD_1~FTFD_3 由 F 导出的所有 TFD 的集合,记作 $\overline{F}^+$,即 $\overline{F}^+=\{X\rightarrow_\mu Y \mid F\vdash_f X\rightarrow_\mu Y\}$.

定义 7(有限属性闭包)[5]. F 是一个有限的 TFD 集,对每个有限属性集 X, X 关于 F 的有限闭包定义为: $\overline{X}^+=\{(B,\mu) \mid X\rightarrow_\mu B\in\overline{F}^+$ 且不存在 $X\rightarrow_\nu B\in\overline{F}^+$使得$\mu\prec\nu\}$.

同传统 FD 集一样,TFD 集也存在冗余现象,为化简 TFD 集同样可定义无冗余覆盖、化简 TFD 集、规范覆盖等概念,对应于传统 FD 集的这些概念的定义见文献[8].

2 相关算法

定义 8(TFD 集的时态类型集). F 是一个 TFD 集,则 F 的时态类型集 $T=\{\nu \mid$ 存在 TFD $X\rightarrow_\nu Y\in F\}$,也称 F 具有时态类型集 T.

定义 9(关联集). μ是任意一个时态类型,F 是一个 TFD 集,那么μ关于 F 的关联集,记作 $\mathrm{Rel}(\mu,F)$,定义为: $\mathrm{Rel}(\mu,F)=\{X\rightarrow_\nu Y \mid X\rightarrow_\nu Y\in F \text{ 且 } \mu\preccurlyeq\nu\}$.

定理 1. 如果 F 是一个具有全序时态类型集的 TFD 集,那么 $F\vDash X\rightarrow_\mu Y$,当且仅当$\pi_\varnothing(\mathrm{Rel}(\mu,F))\vDash X\rightarrow Y$.

证明:(充分性)设$\pi_\varnothing(\mathrm{Rel}(\mu,F))\vDash X\rightarrow Y$,则 $X\rightarrow Y$ 能由$\pi_\varnothing(\mathrm{Rel}(\mu,F)$用 Armstrong[7]公理导出,只需证明 $X\rightarrow_\mu Y$ 能由 F 用前面描述的 4 个推导规则 TFD_1~TFD_4 导出即可.下面我们对推导步数 i 作归纳证明.

初始:当 $i=1$ 时,或者 $Y\subseteq X$ 或者 $X\rightarrow Y\in\pi_\varnothing(\mathrm{Rel}(\mu,F))$.如果 $Y\subseteq X$,由 TFD_1 显然有 $\mathrm{Rel}(\mu,F)\vDash X\rightarrow_\mu Y$;如果 $X\rightarrow Y\in\pi_\varnothing(\mathrm{Rel}(\mu,F))$,由定义 8 必存在 $X\rightarrow_\nu Y\in F$,满足$\mu\preccurlyeq\nu$,根据 TFD_4 有 $F\vDash X\rightarrow_\mu Y$ 成立;

假设:当 $i<n$ 时,即对任意 FD $Z\rightarrow W$,能由$\pi_\varnothing(\mathrm{Rel}(\mu,F))$用 Armstrong 公理小于 n 步导出时,一定有 $F\vDash Z\rightarrow_\mu W$.下面我们来证明归纳步.

归纳:设 $X\rightarrow Y$ 由$\pi_\varnothing(\mathrm{Rel}(\mu,F))$用 Armstrong 公理用 n 步导出,则有下面两种情况:

情形 1:利用 Armstrong 公理用 $n-1$ 步导出 $X\rightarrow V$,然后第 n 步利用增广规则导出 $X\rightarrow Y$,根据归纳假设,有 $F\vDash X\rightarrow_\mu V$,再由 TFD_2 可得 $F\vDash X\rightarrow_\mu Y$;

情形 2:利用 Armstrong 公理用 $n-1$ 步导出 $X\rightarrow V$ 和 $V\rightarrow Y$,然后第 n 步利用传递规则导出 $X\rightarrow Y$,那么由归纳

假设有 $F\vDash X\rightarrow_{\mu}V$ 且 $F\vDash V\rightarrow_{\mu}Y$.根据 TFD_3 可得 $F\vDash X\rightarrow_{\mu}Y$.

上面已经考虑到 $X\rightarrow Y$ 由 $\pi_{\varnothing}(\mathrm{Rel}(\mu,F))$ 用 Armstrong 公理用 n 步导出 $X\rightarrow Y$ 的所有情形,归纳步得证.

(必要性)对于任意被 F 所逻辑蕴含的 $X\rightarrow_{\mu}Y$,只需对 F 利用推导规则 TFD_1~TFD_4 导出 $X\rightarrow_{\mu}Y$ 的步数 i 进行归纳证明.

初始:当 i=1 时,或者 $Y\subseteq X,\mu=\mu_{\mathrm{Top}}$ 或者 $X\rightarrow_{\mu}Y\in F$.对于前一种情况,根据 Armstrong 公理的自反规则,显然有 $\pi_{\varnothing}(\mathrm{Rel}(\mu,F))\vDash X\rightarrow Y$;而对于后者,由定义 9 可知,$X\rightarrow Y\in\pi_{\varnothing}(\mathrm{Rel}(\mu,F))$,显然 $\pi_{\varnothing}(\mathrm{Rel}(\mu,F))\vDash X\rightarrow Y$ 成立.初始步成立.

假设:当 $i<n$ 时,即对任意 TFD $Z\rightarrow_{\nu}W$,能由 F 用 TFD_1~TFD_4 小于 n 步导出时,一定有 $\pi_{\varnothing}(\mathrm{Rel}(\nu,F))\vDash Z\rightarrow W$.下面我们来证明归纳步.

归纳:设 $X\rightarrow_{\mu}Y$ 需要由 F 利用 TFD_1~TFD_4 用 n 步导出,则有下列 3 种情形:

情形 1:F 用 $n-1$ 步导出 $X\rightarrow_{\mu}V$,然后第 n 步利用 TFD_2 导出 $X\rightarrow_{\mu}Y$.由归纳假设一定有 $\pi_{\varnothing}(\mathrm{Rel}(\mu,F))\vDash X\rightarrow V$,由 Armstrong 公理的增广规则可得 $\pi_{\varnothing}(\mathrm{Rel}(\mu,F))\vDash X\rightarrow Y$;

情形 2:F 用 $n-1$ 步导出 $X\rightarrow_{\mu}V$ 和 $V\rightarrow_{\mu}Y$,在第 n 步利用 TFD_3 导出 $X\rightarrow_{\mu}Y$.由归纳假设有 $\pi_{\varnothing}(\mathrm{Rel}(\mu,F))\vDash X\rightarrow V$ 和 $\pi_{\varnothing}(\mathrm{Rel}(\mu,F))\vDash V\rightarrow Y$,根据 Armstrong 公理的传递规则可得 $\pi_{\varnothing}(\mathrm{Rel}(\mu,F))\vDash X\rightarrow Y$;

情形 3:F 用 $n-1$ 步导出 $X\rightarrow_{\mu_1}Y,X\rightarrow_{\mu_2}Y,\ldots,X\rightarrow_{\mu_k}Y$,满足 $\mu\preccurlyeq_C\{\mu_1,\mu_2,\ldots,\mu_k\}$,然后第 n 步利用 TFD_4 导出 $X\rightarrow_{\mu}Y$.由归纳假设一定有 $\pi_{\varnothing}(\mathrm{Rel}(\mu_j,F))\vDash X\rightarrow Y(j=1,2,\ldots,k)$.现设 F 的时态类型集为 T,则由定义 9 必存在 $\mu_j'\in T$,使得 $\mu_j\preccurlyeq\mu_j'$ 并且 $\mathrm{Rel}(\mu_j,F)=\mathrm{Rel}(\mu_j',F)(j=1,2,\ldots,k)$.于是存在 $\mu_l'\in T(1\le l\le k)$,使得对所有 $j\neq l,1\le j\le k,\mu_j'\preccurlyeq\mu_l'$.显然 $\mu\preccurlyeq\mu_l'$,则有 $\mathrm{Rel}(\mu_j',F)\subseteq\mathrm{Rel}(\mu,F)$,于是可得 $\pi_{\varnothing}(\mathrm{Rel}(\mu,F))\vDash X\rightarrow Y$.

上面已考虑到 F 用 n 步导出 $X\rightarrow_{\mu}Y$ 的所有情形,归纳步得证. □

推论 1. 若(R,μ)是一个时态模式,F 是仅包含 R 中属性的 TFD 集,并且 F 的时态类型集是全序的,那么 X 是(R,μ)关于 F 的时态码,当且仅当 X 是 R 关于 $\pi_{\varnothing}(\mathrm{Rel}(\mu,F))$的码.

通过上面的讨论,对于具有全序时态类型集的 TFD 集,可以利用传统 FD 集的有关算法来解决 TFD 集的一些问题.下面首先给出成员籍的算法.

算法 1. 成员籍的算法.

输入:一个具有全序时态类型集的 TFD 集 F 和一个 TFD $X\rightarrow_{\mu}Y$;

输出:如果 $F\vDash X\rightarrow_{\mu}Y$,输出 True,否则输出 False.

```
TO_MEMBERSHIP(F,X→μY)
begin
(1) D =π∅(Rel(μ,F));
(2) if MEMBERSHIP(D,X→Y) then
        return(True);
    else
        return(False);
end
```

定理 2. 算法 TO_MEMBERSHIP 正确判断了一个给定 TFD $X\rightarrow_{\mu}Y$ 是否被 TFD 集 F 所逻辑蕴涵,其时间复杂性是 $O(n+p)$级的.其中 n 表示 F 涉及到的属性的个数,p 表示 F 中 TFD 的个数.

证明:这里用到了传统 FD 的成员籍算法 MEMBERSHIP[8].由于 MEMBERSHIP 算法是正确的,显然 TO_MEMBERSHIP 算法是可终止的.根据定理 1 直接可以证明算法是正确的.对于复杂性,算法的第(1)步至多是 $O(p)$级的,而 MEMBERSHIP 算法的时间复杂性为 $O(n)$级[8],因此算法总的时间复杂度是 $O(n+p)$级. □

有了 TO_MEMBERSHIP 算法,只要用其替换文献[8]中 NONREDUN 和 KEYFINDING 算法中的 MEMBERSHIP 算法即可得到 TFD 集的无冗余覆盖和时态码的算法,本文定义这两个算法分别为 TO-_NONREDUN 和 TO_KEYFIND.同样,用算法 TO_MEMBERSHIP 和 TO_NONREDUN 分别替换文献[8]中算法 CANONICAL 用到的算法 MEMBERSHIP 和 NONREDUN 也可以得到求具有全序时态类型集的 TFD 集规范覆盖的算法 TO_CANONICAL.这些算法的正确性证明过程见文献[8],从中不难分析出算法 TO_NONREDUN, TO_CANONICAL 及 TO_KEYFIND 的时间复杂度分别为 $O(p(n+p))$,$O(n^2p^2)$和 $O(n(n+p))$级(n 表示 TFD 集涉及到的属性的个数,p 表示 TFD 集中的 TFD 个数).

同传统 FD 集一样,对于 TFD 集来说,求解有限属性闭包的算法是非常重要的.在讨论该算法之前,先给出几个重要的引理.

引理 1. 给定具有全序时态类型集 T 的 TFD 集 F,如果 $F\vdash_f X\rightarrow_\mu Y,Y\not\subseteq X$,一定有$\mu\in T$.

证明:对于任一对时态类型μ_1 和μ_2,满足$\mu_1\preccurlyeq\mu_2$,都有μ_1=glb(μ_1,μ_2).只需对 F 应用 $FTFD_1$~$FTFD_3$ 有限导出 $X\rightarrow_\mu Y$ 的步数进行归纳证明即可. □

引理 2. 对于给定 TFD 集 F,若 $F\vdash_f X\rightarrow_\mu Y$,则一定有$\pi_\varnothing(\mathrm{Rel}(\mu,F))\vDash X\rightarrow Y$.

证明:类似于定理 1,根据 Armstrong 公理,只需对 F 应用 $FTFD_1$~$FTFD_3$ 有限导出 $X\rightarrow_\mu Y$ 的步数进行归纳证明即可. □

引理 3. 对于给定 TFD 集 F,若$\pi_\varnothing(\mathrm{Rel}(\mu,F))\vDash X\rightarrow Y$,则一定存在$\nu,\mu\preccurlyeq\nu$,使得 $F\vdash_f X\rightarrow_\nu Y$.

证明:根据定理 1,只需对$\pi_\varnothing(\mathrm{Rel}(\mu,F))$应用 Armstrong 公理导出 $X\rightarrow Y$ 的步数进行归纳证明即可. □

引理 4. 给定具有全序时态类型集的 TFD 集 F,如果 $F\vDash X\rightarrow_\mu A$,且不存在$\nu,\mu\prec\nu$,使得 $F\vDash X\rightarrow_\nu A$,那么一定有$(A,\mu)\in\overline{X}^+$.

证明:对于任意一个 TFD $X\rightarrow_\nu Y$,如果 $F\vdash_f X\rightarrow_\nu Y$,一定有 $F\vDash X\rightarrow_\nu Y$,因此只需证明 $F\vdash_f X\rightarrow_\mu A$ 即可.根据定理 1 可得$\pi_\varnothing(\mathrm{Rel}(\mu_i,F)\vDash X\rightarrow A$;根据已知条件,由引理 3 可得 $F\vdash_f X\rightarrow_\mu A$ 成立. □

算法 2. 有限属性闭包算法.

输入:属性集 X,具有全序时态类型集的 TFD 集 F;

输出:X 关于 F 的有限属性闭包 FCLOS.

TO_FCLOSURE(X,F)

begin

(1) $T=\{\mu|$存在 TFD $Z\rightarrow W\in F\}$;

  FCLOS=$\{(A,\mu_{\mathrm{Top}})|A\in X\}$;

(2) 对 T 中的所有时态类型进行排序,形成全序序列$\mu_1,\mu_2,\ldots,\mu_k(k=|T|)$,使得对任何 $1\le i<j\le k$,有$\mu_i\preccurlyeq\mu_j$;

(3) for i=1 to k do

  [S=LINECLOSURE(X,$\pi_\varnothing(\mathrm{Rel}(\mu_i,F))$);

   $X_i=\{(A,\mu_i)|A\in S-X\}$;]

(4) for i=k downto 2 do

  for 每个$(A,\mu)\in X_i$ do

   if 存在$(A,\nu)\in X_{i-1}$ then

    $X_{i-1}=(X_{i-1}-\{(A,\nu)\})\cup\{(A,\mu)\}$;

(5) FCLOS=FCLOS$\cup X_1$;

  return(FCLOS);

end

定理 3. 算法 TO_FCLOSURE 正确求出了属性集 X 关于 F 的有限属性闭包 FCLOS,其时间复杂性为 $O(k^2+k(n-h)^2+p+h)$级.其中 n 和 p 的含义同定理 2,k 和 h 分别表示 F 的时态类型集所包含的时态类型的个数及 X 所包含的属性个数.

证明:算法第(3)步中调用了在传统函数依赖下求属性集闭包的算法 LINECLOSURE[8],由于该算法是正确的[8],并且算法中仅涉及到了 for 循环,显然算法是可终止的.对于正确性只需证明:(i) 算法结束时,对于 FCLOS 中的每一个元素(A,μ),一定有$(A,\mu)\in\overline{X}^+$;(ii) 每一个元素$(A,\mu)\in\overline{X}^+$,一定被算法在某步加到 FCLOS 中.对于(i),可以看到 FCLOS 中的任何元素(A,μ),要么满足 $A\in X$ 且$\mu=\mu_{\mathrm{Top}}$,要么是某个 $X_i(1\le i\le k)$中的元素,并且对于任何 $X_j(i<j\le k),(A,\mu)\notin X_j$.对于前者,显然$(A,\mu)\in\overline{X}^+$;对于后者,根据定理 1,一定有 $F\vDash X\rightarrow_\mu A$,且不存在$\nu,\mu\prec\nu$,使得 $F\vDash X\rightarrow_\nu A$.根据引理 4 可得$(A,\mu)\in\overline{X}^+$.对于(ii),如果 $A\in X$,除非算法在第(1)步就将(A,μ_{Top})加到 FCLOS 中,否则由

于 F 具有全序时态类型集,根据引理 1 及引理 2,一定存在某个$\mu_i(1\le i\le k)$,$\mu=\mu_i$,使得$\pi_\varnothing(\mathrm{Rel}(\mu_i,F)\vDash X\to A$.再由引理 3,一定不存在$\mu_j(1\le i<j\le k)$,$\mu_i\prec\mu_j$,使得$\pi_\varnothing(\mathrm{Rel}(\mu_j,F)\vDash X\to A$.因此在算法的第(3)步将$(A,\mu)$加到 $X_i(1\le i\le k)$中,并且对于任何 $X_j(1\le i<j\le k)$,(A,μ)都不会被加到 X_j 中.由于$\mu_1\preccurlyeq\mu_i$,显然(A,μ_1)在 X_1 中.最终在算法的第(4)步将(A,μ)加到 X_1 中,进而在第(5)步将其加入到 FCLOS 中.

下面讨论算法的时间花费.显然,算法第(1)步的花费为 $O(p+h)$;算法第(2)步的花费超不过 $O(k^2)$级;算法第(3)步中使用的算法 LINECLOSURE[8]的时间复杂性是 $O(n)$,for 循环次数为 k,因此算法第(3)步的花费是 $O(kn)$级;算法第(4)步中,每个 $X_i(1\le i\le k)$中至多有 $n-h$ 个元素,故该步总的花费不会超过 $O(k(n-h)^2)$级.综上所述,算法总的时间花费为 $O(p+h)+O(k^2)+O(kn)+O(k(n-h)^2)=O(k^2+k(n-h)^2+p+h)$级. □

例 4:给定 TFD 集$\{AB\to_{\mathrm{Year}}CD,DE\to_{\mathrm{Month}}G,CE\to_{\mathrm{Day}}BF,BC\to_{\mathrm{Year}}E\}$,应用算法 TO_FCLOSURE 求 $\overline{AB}^{+}$.

算法第(1)步求得 $T=\{\mathrm{Year},\mathrm{Month},\mathrm{Day}\}$,$\mathrm{FCLOS}=\{(A,\mu_{\mathrm{Top}}),(B,\mu_{\mathrm{Top}})\}$;

算法第(2)步得到偏序序列 Day,Month,Year;

算法第(3)步得到 $X_1=\{(C,\mathrm{Day}),(D,\mathrm{Day}),(E,\mathrm{Day}),(F,\mathrm{Day}),(G,\mathrm{Day})\}$,$X_2=\{(C,\mathrm{Month}),(D,\mathrm{Month}),(E,\mathrm{Month}),(G,\mathrm{Month})\}$,$X_3=\{(C,\mathrm{Year}),(D,\mathrm{Year}),(E,\mathrm{Year})\}$;

算法第(4)步得到 $X_1=\{(C,\mathrm{Year}),(D,\mathrm{Year}),(E,\mathrm{Year}),(F,\mathrm{Day}),(G,\mathrm{Month})\}$;

最终在第(5)步得到 $\overline{AB}^{+}$,即 $\mathrm{FCLOS}=\{(A,\mu_{\mathrm{Top}}),(B,\mu_{\mathrm{Top}}),(C,\mathrm{Year}),(D,\mathrm{Year}),(E,\mathrm{Year}),(F,\mathrm{Day}),(G,\mathrm{Month})\}$.

3 结束语

通常,数据库中用秒(second)、分(minute)、小时(hour)、天(day)、月(month)和年(year)等时间粒度来记录事件发生或数据有效的时间,而由这些粒度组成的时态类型集关于“细于”关系是全序的.因此本文研究的算法对时态数据库设计来说具有普遍的意义.文中给出的成员籍、TFD 集化简及有限属性闭包等算法是寻找有效的设计满足 T3NF 或 TBCNF 的时态数据库模式算法的前提和基础.后续的主要工作有:(1) 利用本文的工作,寻找有效的时态模式的 T3NF 和 TBCNF 的分解算法;(2) 具有非全序时态类型集的 TFD 集的相关算法研究;(3) 时态类型有关操作的计算机实现研究等.

References:

[1] Jensen CS., Clifford J. A glossary of temporal database concepts. ACM SIGMOD Record, 1994,23(1):52~64.

[2] Jensen CS, Snodgrass RT. Semantics of time-varying information. Information Systems, 1996,21(4):311~352.

[3] Jensen CS, Snodgrass RT, Soo MD. Extending existing dependency theory to temporal databases. IEEE Transactions on Knowledge and Data Engineering, 1996,8(4):563~582.

[4] Wijsen J. Design of temporal relational databases based dynamic and temporal functional dependencies. In: Clifford S, Tuzhlin A, eds. Proceedings of the International Workshop on Recent Advances in Temporal Databases. Berlin: Springer-Verlag, 1995. 61~76.

[5] Wang XS, Bettini C., Jajodia S. Logical design for temporal databases with multiple. ACM Transactions on Database Systems, 1997,22(2):115~170.

[6] Wijsen J. Temporal FDs on complex objects. ACM Transactions on Database Systems, 1999,24(1):127~176.

[7] Ullman JD. Principles of Database and Knowledge-Base Systems. Rockville: Computer Science Press, 1988.

[8] Shi BL, Hen JC, Cui J. The Theory and Application of Relational Databases. Zhengzhou: Henan Science and Technology Publishing House, 1989. 359~382 (in Chinese).

附中文参考文献:

[8] 施伯乐,何继潮,崔靖.关系数据库理论及应用.郑州:河南科学技术出版社,1989.359~382.